Министерство науки, высшей школы и технической политики России.

Комитет по высшей школе.

Рыбинский авиационный технологический институт.

Решение головоломки "Ханойские башни" на машине Тьюринга

И.Басов

Рыбинск 1993 г.

В этой статье приводится машина Тьюринга для решения головоломки "ханойские башни". Несмотря на кажущуюся сложность задачи, идея проста и мы надеемся, что она будет интересна не только студентам - программистам, но и дрругим читателям.

Для полноты изложения приведем разъяснение понятий "ханойские башни" и "машина Тьюринга".

Ханойские башни.

Эта головоломка известна уже довольно давно. Ее автором принято считать французского математика Э. Люка, создавшего ее на основе древних легенд.

Приведем одну из таких легенд:

В одном из храмов Юго - Восточной Азии находятся три вертикальных стержня, на которые нанизаны 64 золотых кольца разного диаметра (рис 1). Некогда бог Вишну поместил все 64 кольца на первый стержень так, что диаметр колец снизу вверх уменьшался, и повелел жрецам переместить башню из колец с первого стержня на второй, соблюдая следующее правило: на каждом шаге можно перенести самое верхнее кольцо с одного из стержней на верх другого стержня при условии, что на каждом из стержней кольца будут сохранять форму башни (т.е. их диаметр снизу вверх уменьшается). С тех пор много тысяч лет жрецы днем и ночью перекладывают кольца. Легенда гласит, что когда все кольца окажутся на третьем стержне, наступит конец света.

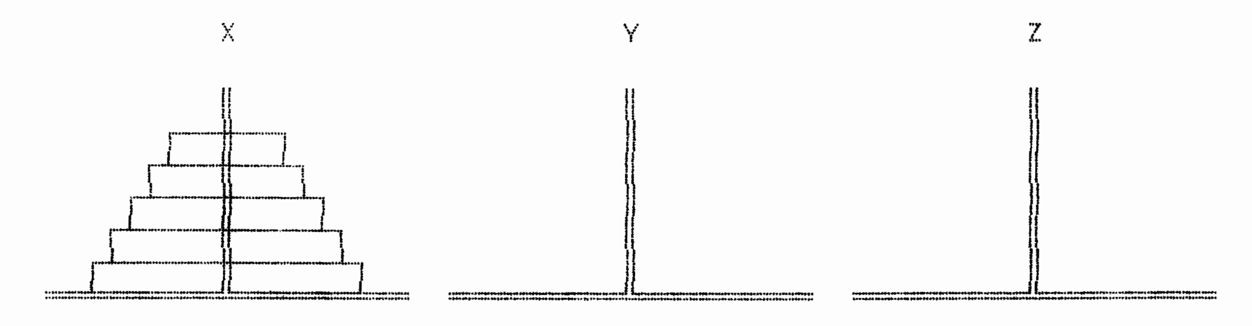


Рис. 1

Классическим стал рекурсивный алгоритм решения этой задачи. Пусть n - количество колец. Предположим, что мы умеем переносить башню из (n-1) колец. Тогда последовательность действий такая:

Если n=1 то надо просто переместить кольцо со стержня x на стержень y.

Если n>1 то следует переместить (n-1) колец со стержня x на стержень z (что мы имеем право сделать по предположению), затем переместить самое нижнее кольцо со стержня x на стержень y, и, наконец, все (n-1) колец со стержня z переносим на стержень y. И все. Задача решена!

Например, на языке программирования Паскаль этот алгоритм реализуется так:

```
Procedure Hanoi(n: integer; x,y,z: char);
Begin
  If n<>0 Then
  Begin
    Hanoi(n-1,x,z,y);
    Writeln('Переместить кольцо со стержня ',x,' на стержень ',y);
    Hanoi(n-1,z,y,x)
  End
End;
```

Очевидно, что для обоснования алгоритма необходимо доказать следующие три утверждения:

1) Программа завершит свою работу при любом начальном числе колец на первом стержне.

2) После завершения работы программы кольца образуют правильную башню на втором стержне.

3) В ходе работы программы не нарушается правило расположения колец: для любой пары колец на любом из стержней диаметр верхнего кольца меньше, чем диаметр нижнего.

Мы эти утверждения доказывать не будем. Отметим только, что доказательство проводится индукцией по числу колец n.

Заметим, что число шагов, требующихся для перемещения башни из n колец равняется

$$U_n = 2^n-1.$$

Доказательство проводится индукцией по числу колец n:

$$U_1 = 2^1-1 = 1$$
$$U_{n+1} = U_n+1+U_n = 2U_n+1 = 2(2^n-1)+1 = 2^{n+1}-2+1 = 2^{n+1}-1$$

Таким образом, для решения задачи с 64 кольцами потребуется число перемещений, равное $2^{64}-1$, или около 10^{20}. Необходимое время - десять миллионов лет на сверхбыстродействующей ЭВМ. Что касается "конца света", то он произойдет по истечении более пяти миллиардов веков, если считать, что одно кольцо перемещается за одну секунду.

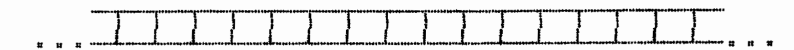
Машина Тьюринга.

Машина Тьюринга была предложена в мае 1936 года в статье "О вычислимых числах с приложением к проблеме разрешения" английским математиком Аланом Мэтисоном Тьюрингом как формализация интуитивного понятия алгоритма.

Машина Тьюринга имеет следующие составные части:

I) Конечный алфавит $A=\{a_\varnothing,a_1,...,a_n\}$, $a_\varnothing$=@ - пустой символ.

II) Внешняя память в виде бесконечной ленты, разбитой на ячейки:



III) Головка.

Головка на каждом такте обозревает одну ячейку. Головка может считать символ и передать его в устройство управления.

Головка может получить из устройства управления символ и записать его в ячейку.

Головка может либо оставаться на месте, либо сдвинуться на ячейку вправо, либо - влево. Алфавит $P=\{\leftarrow,\downarrow,\rightarrow\}$ содержит символы, идентифицирующие эти действия.

IV) Устройство управления.

В каждом такте устройство управления находится в одном из возможных состояний алфавита $Q=\{q_\varnothing,q_1,...,q_m\}$, $q_\varnothing=q_{нач}$, $q_m=q_{кон}$.

В каждом такте, находясь в состоянии q_j, устройство управления делает:

1) получает информацию о считываемом головкой символе a_i;

2) вырабатывает команду, которую передает головке на исполнение.

3) переходит в новое состояние q_r.

Команда содержит:

а) Символ a_r алфавита A, который головка должна записать в ячейку;

б) Символ p_r алфавита P, определяющий движение головки;

в) Символ q_r алфавита Q, определяющий состояние, в которое должно перейти устройство управления.

Примечание: все три алфавита A,P,Q взаимно не пересекаются, т.е.
$A \& P = \varnothing$, $A \& Q = \varnothing$, $Q \& P = \varnothing$.

Итак, по заданному набору символов на ленте наша машина должна выдать результат и остановиться.

Договоримся, что в начальный момент в поле зрения всякой машины Тьюринга находится самая левая непустая ячейка ленты.

Машина Тьюринга полностью задается таблицей 1.

Таблица 1.

состояние	обозреваемый символ				
	a_0	...	a_i	...	a_n
q_0		...		...	
::::	::::	::::	::::	::::	::::
q_j		...	a_r p_r q_r	...	
::::	::::	::::	::::	::::	::::
q_m		...		...	

Где

a_i — обозреваемый в данный момент символ;
q_j — состояние, в котором находится машина;
a_r, p_r, q_r — составнае части команды.

Договоримся использовать следующие сокращения:

1) Если $q_r = q_j$ то q_r в клеточке не пишется;
2) если $a_r = a_i$ то a_r в клеточке не пишется;
3) если $p_r = \downarrow$ то p_r в клеточке не пишется;
4) $q_{кон} = !$;
5) если вариант a_i, q_j при работе машины не встречается, то соответствующую клеточку будем оставлять пустой.

Теперь можно перейти к цели нашего повествования — составлению машины Тьюринга для решения задачи "ханойские башни".

Во-первых введем представление "Ханойских башен", удобное для машины Тьюринга. Пусть мы имеем три типа символов: x,y,z. Запишем в несколько подряд идущих ячеек ленты один и тот же символ, например — x. Это будет означать, что все кольца находятся на стержне x.

Установим следующую нумерацию:

символы нумеруются справа налево, кольца — от меньшего к большему. Самый правый символ и наименьшее кольцо имеют номер 1. Далее по порядку.

Если мы заменим, например, какой-нибудь символ x на символ y, то будем считать, что перенесли соответствующее кольцо со стержня x на стержень y. Задача будет решена, когда все символы x будут заменены на символы y. При замене нужно руководствоваться следующими правилами:

1. Заменять по одному символу.
Это условие - аналог того,что нужно переносить с одного стержня на другой только по одному кольцу (выполняется самим устройством машины);
2. Заменять некоторый символ можно тогда,когда правее него нет таких символов.
Это условие - аналог того,что можно брать со стержня только самое верхнее кольцо;
3. Заменять символ можно только на тот символ,какого нет среди символов,расположенных правее заменяемого.
Это условие - аналог того,что нельзя класть кольцо большего диаметра на кольцо меньшего диаметра;

Во вторых нужно выбрать нерекурсивный алгоритм для решения этой задачи. Очевидно,что описанный выше рекурсивный алгоритм не подходит для нашего случая, так как его реализация на машине Тьюринга весьма громоздка и требует дополнительного места на ленте для стека,чтобы запомнить конфигурацию основного слова перед очередными рекурсивными вызовами.

Присмотримся повнимательнее к процессу переноса колец при решении задачи. Начнем переносить кольца в соответствии с алгоритмом и будем записывать номера переносимых колец:

1 2 1 3 1 2 1 4 1 2 1 3 1 2 1 5 ...

Любопытно,что каждый нечетный перенос - это перенос первого кольца. Еще одно наблюдение : расположим стержни в вершинах равностороннего треугольника и понаблюдаем за движением первого кольца. Оказывается,что оно движется по кругу в одну и ту же сторону. При этом ,если начальное количество колец четно, то по часовой стрелке, а если нечетно - против. Эти наблюдения позволили в 1961 году (не знаю кому) сформулировать следующий алгоритм переноса колец:

- сначала переносим первое кольцо,потом не первое и т.д.,причем первое кольцо переносится в одном направлении: по часовой стрелке в случае четного количества колец и против часовой стрелки, в случае их нечетного количества.

Заметим,что указание "переносится не первое кольцо" полностью определяеткакое кольцо и куда следует переносить в данный момент, поскольку меньшее кольцо всегда кладется на большее.

Как будет выглядеть только что сформулированный алгоритм,применительно к нашему представлению задачи для машины Тьюринга ?

Сначала обговорим некоторые детали. Для того,чтобы машина была универсальной, во входном слове нужно задать помимо начальных n одинаковых символов,характеризующих стержень - источник и n колец на нем, символ, на который нужно заменить первоначальный символ. Этот символ будет олицетворять стержень - приемник.

Пусть,например, входное слово будет таким:

"y*xxx"

Это означает,что символы 'x' нужно заменить на 'y' или перенести пирамиду из трех колец со стержня x на стержень y. В начальный момент головка находится на самом левом не пустом символе,т.е. - на 'y'. Далее мы должны запомнить в одном из состояний символ - приемник,стереть первые два символа (в данном случае 'y' и '*'), определить четное или нечетное количество символов - источников (для этого нужно дойти до конца слова) и заменять символы в ячейках в соответствии с алгоритмом.

В конце концов должно получится такое выходное слово : "yyy". Это значит, что все кольца со стержня x перенесены на стержень y.

Разберем подробнее алгоритм. Указание "переносим первое кольцо" равносильно - "заменяем символ, расположенный в самой правой непустой ячейке" (она значится под номером 1). Рассмотрим что значит "переносим не первое кольцо" (или, что тоже самое "заменяем символ, расположенный не в самой правой непустой ячейке"). Как найти этот символ? На какой символ его заменить? Ответ прост. Если вспомнить описанные выше правила замены символов, то его можно сформулировать так: нужно двигаться справа налево до тех пор, пока не встретится символ, отличный от самого правого (и от всех символов, расположенных правее него). Разумеется, когда мы говорим "заменить символ", то подразумеваем, что его нужно заменить на символ, отличный от исходного. Таким образом, все эти указания полностью определяют какой символ и на какой следует заменить на каждом шаге.

Осталось выяснить в каких случаях первое "кольцо" движется по часовой стрелке, а в каких против. Разобьем задачу о ханойской башне на два класса в зависимости от того, с какого стержня и на какой следует переносить кольца (какие символы и на какие заменять):

Класс 1

"x" заменить на "y" или "y" заменить на "z" или "z" заменить на "x". Схематически это изображено на рис. 2.

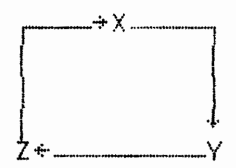


Рис. 2

Класс 2

"x" заменить на "z" или "z" заменить на "y" или "y" заменить на "x". Схематически это изображено на рис. 3.

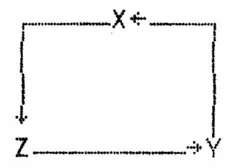


Рис. 3

В обоих случаях, если число символов нечетно, то порядок замены первого совпадает с направлением стрелок, а если четно, то - противоположен.

В таблице 2 приведена машина Тьюринга для решения нашей задачи. Для этой машины A={x,y,z,■,@}, Q={q_0,q_1,...,q_{15},!}

Таблица 2

номер блока	состо- яние машины	обозреваемый символ				
		x	y	z	■	@
1	q_0	@ → q_1	@ → q_2	@ → q_3		
2	q_1	!	→ q_4	→ q_5	@ →	
	q_2	→ q_5	!	→ q_4	@ →	
	q_3	→ q_4	→ q_5	!	@ →	
3	q_4	→ q_5	→ q_5	→ q_5		← q_{11}
	q_5	→ q_4	→ q_4	→ q_4		← q_6
4	q_6	y ← q_7	z ← q_8	x ← q_9		
	q_7	x → q_{10}	←	x → q_{10}		→ !
	q_8	y → q_{10}	x → q_{10}	←		→ !
	q_9	←	z → q_{10}	y → q_{10}		→ !
	q_{10}	→	→	→		← q_6
5	q_{11}	z ← q_{12}	x ← q_{13}	y ← q_{14}		
	q_{12}	y → q_{15}	x → q_{15}	←		→ !
	q_{13}	←	z → q_{15}	y → q_{15}		→ !
	q_{14}	z → q_{15}	←	x → q_{15}		→ !
	q_{15}	→	→	→		← q_{11}

Эта программа имеет простую линейную структуру и выполняется "сверху вниз". Разберем что выполняют отдельные ее блоки:

блок 1 – выбор пути решения в зависимости от того, на какой стержень нужно переместить пирамиду;

блок 2 – выбор пути решения в зависимости от того, на каком стержне находилась пирамида вначале;

блок 3 – выбор пути решения в зависимости от числа колец (четное или нечетное):

блок 4 - решение задачи, если она относится к классу 1 (рис. 2) и число колец нечетно, или - к классу 2 (рис. 3) и число колец четно;

блок 5 - решение задачи, если она относится к классу 2 (рис. 3) и число колец нечетно, или - к классу 1 (рис. 2) и число колец четно.

Нам представляется, что приведенная машина Тьюринга наиболее эффективна, т.е. она призводит работу за наименьшее число тактов (понятие такта дано выше), т.к. каждый такт сопровождается перемещением головки, а головка двигается в строгом соответствии с алгоритмом. Найдем оценку числа тактов когда число колец равно n.

Первые два такта (блок 1 и блок 2) тратятся на то, чтобы пройти вправо на две ячейки, попутно стирая символы, записанные в них. Вконце второго такта головка останавливается на самом левом символе - источнике. Далее еще за n тактов (блок 3) головка достигает первого пустого символа @ справа от слова (эти такты уходят на распознавание: четно или нечетно число символов).

Теперь определим колличество тактов, необходимое для основной работы - замены символов (блок 4 или блок 5). Заметим, что порядковый номер кольца, а в данном случае символа, четко связан с колличеством тактов, необходимых, чтобы его обработать. Для удобства в качестве точки отсчета возмем самый первый справа от слова пустой символ и назовем его нулевым. Определим понятие обработки символов. Символы обрабатываются попарно (первый и не-первый), кроме самого последнего шага обработки, когда первому символу недостается пары, т.к. его замена завершает процесс основной работы. Процесс обработки пары символов начинается с того, что головка передвигается с нулевого символа на первый, заменяет его, передвигается на k-ый символ, заменяет его и возвращается к нулевому символу. Для этого потребуется (2k-2) такта.

Обозначим V_n - общее число тактов, которые должна совершить машина Тьюринга, чтобы заменить n символов. Зная, что формула имеет рекуррентную структуру, запишем ее так:

$$V_{n+1} = V_n+2n+V_n = 2V_n+2n$$

или, перенеся $2V_n$ в левую часть,

$$V_{n+1}-2V_n = 2n.$$

Решим это неоднородное рекуррентное уравнение при начальном условии $V_1 = 2$. Для решения воспользуемся стандартным методом, применяемым к уравнениям такого типа (Т.Фудзисава, Т.Касами - Математика для радиоинженеров. Теория дискретных структур.- М.: Радио и связь, 1984).

Данное уравнение имеет характеристический многочлен $g(z) = z-2$. Так как $g(1) = -1 \neq 0$, то частным решением будет многочлен первой степени. Подставляя $V_n = b_0+b_1n$ в уравнение, получаем

$$[b_0+b_1(n+1)]-2[b_0+b_1n] = -b_0+b_1-b_1n = 2n.$$

Отсюда следует, что $b_0 = b_1 = -2$. Найдя частное решение, положим

Из начальных условий находим $V_1 = -2-2*1+c*2^1 = -4+2c$, откуда получаем c=3.

Таким образом,

$$V_n = 3*2^n-2n-2.$$

После того, как основная работа совершена, т.е. все символы - источники заменены на символы - приемники, головка движется влево до первого пустого символа @. На это уходит (n-1) тактов. После этого головка еще за один такт сдвигается вправо на начало слова. И все. Работа завершена.

Теперь подсчитаем общее колличество тактов, совершенных за все время работы машины:

$$S_n = 2+n+(3*2^n-2n-2)+(n-1)+1 = 3*2^n$$

$$\boxed{S_n = 3*2^n}$$

Проверим полученный результат на нашем примере, когда слово - вход есть "y▪xxx". Здесь n=3 и количество тактов равно $S_3 = 3*2^3 = 24$. Работа машины приведена в таблице 3.

Таблица 3.

номер такта	положение головки на ленте и состояние устройства упр.						
0		y	■	x	x	x	
		q_0					
1			■	x	x	x	
			q_2				
2				x	x	x	
				q_2			
3				x	x	x	
					q_3		
4				x	x	x	
						q_4	
5				x	x	x	
							q_5
6				x	x	x	
						q_6	
7				x	x	y	
					q_7		
8				x	z	y	
						q_{10}	
9				x	z	y	
							q_{10}
10				x	z	y	
						q_6	
11				x	z	z	
					q_8		

12				x	z	z	
				q_8			
13				y	z	z	
					q_{10}		
14				y	z	z	
						q_{10}	
15				y	z	z	
							q_{10}
16				y	z	z	
						q_6	
17				y	z	x	
					q_9		
18				y	y	x	
						q_{10}	
19				y	y	x	
							q_{10}
20				y	y	x	
						q_6	
21				y	y	y	
					q_7		
22				y	y	y	
				q_7			
23				y	y	y	
			q_7				
24				y	y	y	
				!			

В случае, когда исходный и конечный стержни совпадают, то ничего перекладывать не нужно и количество колец, подлежащих перекладыванию можно положить равным нулю (n=0). В этом случае полученная формула для количества тактов также остается в силе:

$$S_0 = 3*2^0 = 3$$

Проверим ее на таком примере: "x■xxx". Работа машины приведена в таблице 4.

Таблица 4.

номер такта	положение головки на ленте и состояние устройства упр.						
0		x	■	x	x	x	
		q_0					
1			■	x	x	x	
			q_1				
2				x	x	x	
				q_1			
3				x	x	x	
				!			

Нерекурсивный алгоритм, по которому может быть построена машина Тьюринга для решения задачи, не единственный. В публикации А.Савина "Ханойская башня" [1] приведен алгоритм М.Федорова. В этом алгоритме башню, которая не учавствует на данном шаге в операции переноса кольца, называют пустой. Алгоритм формулируется очень коротко:

в процессе перекладывания пустая башня должна двигаться по кругу в одном направлении: по часовой стрелке, если число дисков нечетно, и против, если их число четно.

Проанализировав движение пустой башни, мы видим, что оно направлено в сторону, противоположную движению первого диска.

В качестве упражнения читателям предлагается самостоятельно внести в построенную машину Тьюринга некоторые изменения с тем, что бы она работала в соответствии с алгоритмом М.Федорова, а также оценить число тактов, необходимое для ее работы.

Рассмотрим расширение понятия ханойских башен – р-башни (распределенные башни), когда в начальный момент все кольца собраны не на одном стержне, а могут быть распределены по всем трем стержням. Требуется собрать все кольца на каком-то одном из стержней (все правила начального положения и перекладывания остаются в силе). Эта задача также имеет решение на машине Тьюринга и входное слово в этом случае может иметь, например, такой вид: "y▪xxzxyz".